Adaptive Scenario Partitioning for Stochastic Bilevel Linear Programs

Carlos Muñoz-Rey[a], Francisco Jara-Moroni[a,*]

[a]School of Industrial Engineering, Faculty of Engineering and Sciences, Diego Portales University, 441 Ejército Avenue, Santiago, 8370074, Metropolitan Region, Chile

ARTICLE INFO

Keywords:
Stochastic bilevel programming
Adaptive partitioning
Scenario aggregation
Basis-aware refinement
Decomposition methods
Computational optimization

ABSTRACT

This paper develops an adaptive scenario partitioning approach for stochastic bilevel linear programs. The method extends the Adaptive Partitioning Method, originally designed for two-stage stochastic programs, to settings in which a leader makes a first-stage decision while anticipating scenario-dependent optimal responses from a follower. The proposed approach solves a sequence of aggregated master problems and refines scenario groups using information from the follower's optimal bases and primal–dual signatures. We also study computational variants based on basis-aware refinement, scenario bundling, and non-trivial initial partitions obtained from K-means clustering and central initialization. The approach is evaluated on stochastic continuous bilevel knapsack instances with up to 3,200 scenarios and compared with an extended reference formulation. The results show that basis-aware refinement and informative initial partitions substantially reduce solution times while keeping objective gaps small relative to the reference model. These findings indicate that adaptive partitioning is a promising computational strategy for large-scale stochastic bilevel optimization.

1. Introduction

Bilevel optimisation provides a mathematical framework for hierarchical decision-making in which a leader selects a decision while anticipating the optimal response of a follower. This modelling paradigm originates from the sequential games introduced by von Stackelberg (1934), where one player commits to an action before the other reacts optimally. Since then, bilevel models have been widely adopted to represent leader–follower interactions in pricing and tariff design, energy planning, network design, investment decisions, and regulatory problems.

Even in their simplest form—linear objectives and constraints at both levels—bilevel programs are notoriously challenging. The feasible region of the leader is implicitly defined by the follower's optimal solution set, which may be nonconvex, disconnected, and highly irregular. As a consequence, the leader's objective function often lacks smoothness and standard regularity properties, and the resulting optimization problem belongs to the class of NP-hard problems. These intrinsic structural difficulties have been extensively analysed in the bilevel optimization literature (see Beck and Schmidt (2021), Dempe (2002)).

A common approach to solving deterministic bilevel linear programs relies on reformulating the follower's problem through its Karush–Kuhn–Tucker (KKT) optimality conditions, yielding single-level mixed-integer linear programs. The classical formulation proposed by Fortuny-Amat and McCarl (1981) remains one of the most widely used exact techniques. However, such reformulations typically assume that all follower-related parameters are known with certainty and quickly become computationally intractable as problem size increases.

In many practical applications, this assumption is unrealistic. The follower's costs, resource availabilities, or technological parameters are often affected by uncertainty, leading to outcomes that vary across future scenarios. Ignoring this uncertainty may cause the leader to select decisions that perform well under a single realisation but are fragile when evaluated across the full distribution of possible outcomes. This observation motivates the study of stochastic bilevel programming (SBLP), where the leader commits to a decision before observing uncertain parameters and optimises an expected objective over scenario-dependent follower responses.

Stochastic bilevel models significantly increase computational complexity. Not only must the follower's problem be solved repeatedly across many scenarios, but uncertainty may also induce scenario-wise changes in the follower's optimal basis, generating abrupt shifts in feasibility and objective values from the leader's perspective. As a result, extensive-form reformulations and standard decomposition techniques often scale poorly when applied directly to large-scale stochastic bilevel problems. While two-stage stochastic programming provides a natural conceptual analogue, the presence of a distinct rational agent at the second level introduces structural challenges that are absent in classical recourse models (see Shapiro, Dentcheva and Ruszczyński (2009)).

Among decomposition-based approaches for stochastic optimization, the Adaptive Partitioning Method (APM) introduced by Song and Luedtke (2015) has proven particularly effective for large-scale two-stage problems with fixed recourse. APM begins with a coarse aggregation of scenarios and iteratively refines only those subsets that contribute most to approximation error, exploiting the piecewise-linear

This work was supported by ANID FONDECYT Iniciación grant No. 11240220.

*Corresponding author
carlos.munoz2@mail.udp.cl (C. Muñoz-Rey); francisco.jara@mail.udp.cl (F. Jara-Moroni)
ORCID(s): 0009-0006-3629-9819 (C. Muñoz-Rey); 0000-0003-3381-3078 (F. Jara-Moroni)

structure induced by dual bases. This idea was later generalised to broader uncertainty structures by Ramirez-Pico and Moreno (2022).

The structural properties that make APM successful in two-stage stochastic programming are especially appealing in the bilevel context. Recent theoretical advances in stochastic linear bilevel programming emphasise the central role of polyhedral geometry and basis-dependent behaviour in determining algorithmic performance (see Muñoz, Salas and Svensson (2025)). These insights suggest that adaptive aggregation based on basis or dual information is not merely a computational heuristic, but a principled mechanism for controlling approximation error while respecting the geometry induced by the follower's optimal responses.

This paper extends adaptive scenario partitioning to stochastic bilevel linear programs and positions it as a computational strategy for reducing the burden of scenario-wise follower optimization. The contribution is threefold. First, we formulate an aggregated master problem that preserves the leader–follower structure while replacing individual scenarios with dynamically updated scenario blocks. Second, we introduce bilevel-aware refinement rules based on the follower's primal–dual information and, in particular, on optimal basis-aware signatures; these rules are designed to separate scenarios only when they induce structurally different follower responses. Third, we assess several acceleration mechanisms—including scenario bundling, K-means initial partitions, and central initialization—to quantify the trade-off between computational speed and agreement with an extended reference formulation.

To evaluate the proposed approach, we conduct an extensive computational study based on the stochastic bilevel knapsack problem, a benchmark that captures the interaction between hierarchical decision-making, combinatorial structure, and uncertainty. This problem has been studied in both deterministic and stochastic settings (see Buchheim, Henke and Irmai (2022)) and provides a controlled environment for assessing scalability, solution quality, and convergence behaviour. Our results show that the adapted APM variants can substantially reduce computation times relative to the extended formulation while maintaining small objective-value gaps across a wide range of instances.

The remainder of this paper is organised as follows. Section 2 reviews related work on bilevel and stochastic optimisation. Section 3 presents the adaptive partitioning algorithm for SBLPs. Section 4 reports numerical results, and Section 5 concludes.

2. Literature Review

This section positions the proposed adaptive scenario partitioning method within three streams of literature: computational bilevel optimization, stochastic bilevel programming, and scenario aggregation for stochastic optimization. The aim is not to provide an exhaustive survey of bilevel optimization, but rather to identify the methodological gap addressed by this paper.

2.1. Bilevel optimization and computational reformulations

Bilevel optimization models hierarchical decision-making problems in which an upper-level decision maker anticipates the optimal response of a lower-level decision maker. The paradigm dates back to Stackelberg competition (von Stackelberg, 1934) and has since become a standard modeling framework for pricing, network design, interdiction, logistics, energy planning, and regulation. Even for linear upper- and lower-level problems, bilevel programs are computationally challenging because the feasible region induced by the follower's optimal reaction is generally nonconvex and may be disconnected (Dempe, 2002). Recent surveys provide updated views of both the modeling scope and algorithmic landscape: Beck, Ljubić and Schmidt (2023b) review bilevel optimization under uncertainty, Kleinert, Labbé, Ljubić and Schmidt (2021) survey mixed-integer programming techniques for deterministic bilevel optimization, and Camacho-Vallejo, Corpus and Villegas (2024) review metaheuristic approaches for bilevel models.

A common computational route for linear bilevel problems replaces the lower-level problem by its optimality conditions and then linearizes complementarity constraints using binary variables and big-M constants, following the classical approach of Fortuny-Amat and McCarl (1981). This reformulation is attractive because it converts the bilevel problem into a single-level mixed-integer model, but it also introduces numerical sensitivity, weak relaxations, and potentially many binary variables. These issues become more severe when uncertainty is represented by many scenarios, because each scenario can induce a distinct lower-level optimality system. Thus, while reformulation-based methods remain an essential baseline, large-scale stochastic bilevel instances require additional structure-exploiting algorithmic ideas.

Stochastic bilevel optimization extends the deterministic leader–follower setting by allowing uncertain parameters to affect either the leader's objective, the follower's objective, the follower's feasible region, or combinations of these components. In the finite-scenario case, the leader typically chooses a here-and-now decision before observing uncertainty, while the follower's response is scenario dependent. This creates a deterministic equivalent with repeated lower-level optimality systems, one per scenario, and therefore combines the nonconvexity of bilevel optimization with the dimensional growth of stochastic programming.

Recent work has clarified important theoretical and structural aspects of this class of problems. Claus (2021) studies continuity issues in risk-averse bilevel stochastic linear programming when the lower-level objective is random, and Claus (2022) establishes existence results for a class of bilevel stochastic linear programs. For the benchmark used in this paper, Buchheim et al. (2022) analyze the stochastic bilevel continuous knapsack problem with uncertain follower objective coefficients. More recently, Muñoz et al. (2025) exploit the polyhedral geometry of stochastic linear bilevel programming and show the relevance of chamber

structures and basis-dependent behavior. These contributions motivate algorithms that do not merely aggregate scenarios statistically, but instead use structural information from the follower's optimal response.

The uncertainty literature also highlights the importance of lower-level instability. When the follower has multiple optimal or near-optimal solutions, small changes in data or solver behavior can lead to different lower-level responses. Besançon, Anjos and Brotcorne (2024) address robust bilevel optimization with near-optimal lower-level solutions, emphasizing that the leader's problem may be sensitive to how lower-level ambiguity is resolved. This observation is directly relevant to the present paper: our basis-aware refinement is designed to distinguish scenarios according to the follower's optimal basis information rather than only by rounded primal solutions, thereby reducing artificial aggregation or fragmentation caused by degeneracy and numerical noise.

Other recent computational work under uncertainty further illustrates the breadth of stochastic and robust bilevel models. Pinzon Ulloa, Frejinger and Gendron (2024) formulate a stochastic bilevel facility-location problem with random-utility-maximizing followers, while García-Vélez, Ruiz-Hernández, Camacho-Vallejo and Díaz (2024) study a bilevel facility delocation problem in competitive network restructuring. Beck, Ljubić and Schmidt (2023a) develop exact methods for discrete Γ-robust interdiction problems with an application to the bilevel knapsack problem, and Lozano and Borrero (2024) propose bilevel optimization methods for combinatorial problems with disruptions and probing. In parallel, the machine-learning literature studies stochastic bilevel optimization through stochastic-gradient methods under smoothness assumptions (Chen, Xiao and Balasubramanian, 2024); this stream is algorithmically distinct from the finite-scenario stochastic linear bilevel setting considered here, but it underscores the growing interest in stochastic hierarchical optimization.

The proposed method builds most directly on adaptive partitioning for two-stage stochastic programming. In two-stage stochastic linear programs with fixed recourse, the Adaptive Partitioning Method of Song and Luedtke (2015) solves a sequence of aggregated master problems and refines scenario groups using dual information. Ramirez-Pico and Moreno (2022) generalize this idea to broader fixed-recourse structures. The central insight is that scenarios need not always be represented individually: if scenarios induce the same relevant dual or basis structure, they can be handled as a block without losing the essential geometry of the recourse function.

Related ideas appear in scenario generation, scenario reduction, quantization, and decomposition for large finite-scenario models. Fairbrother, Turner and Wallace (2022) argue that scenario generation should be problem driven rather than purely distributional, especially when risk measures or tail behavior matter. Bounitsis, Papageorgiou and Charitopoulos (2022) propose data-driven scenario generation for two-stage stochastic programming using clustering and moment information. Forcier, Gaubert and Leclère (2024) study exact quantization of multistage stochastic linear problems, linking finite scenario representations to polyhedral structure. Salo, Andelmin and Oliveira (2022) also emphasize finite-scenario representations for mixed-integer multistage optimization under uncertainty. Roland, Forel and Vidal (2025) propose the adaptive partition method to chance constrained problems with finite support. These works support the view that effective scenario representations should reflect the optimization problem's structure, not only the statistical distribution of uncertainty.

Computational decomposition methods provide a complementary perspective on scalability. Rahmaniani, Crainic, Gendreau and Rei (2024) develop an asynchronous parallel Benders decomposition method for stochastic network design, Mahéo, Belières, Adulyasak and Cordeau (2024) propose a unified Branch-and-Benders-Cut framework for two-stage stochastic mixed-integer programs, and Zhang, Mazzi, McKinnon, Garcia Nava and Tomasgard (2024) introduce stabilized Benders decomposition with adaptive oracles for large-scale stochastic programming. Kermani, Cordeau and Jans (2024) develop a progressive-hedging-based matheuristic for a stochastic production-routing problem. Although these methods address single-level stochastic programs, they reinforce the computational motivation for decomposition, aggregation, stabilization, and scenario-wise acceleration in large-scale stochastic optimization.

Our contribution differs from this literature in two ways. First, the second-stage model is not a recourse problem controlled by the same decision maker, but a follower optimization problem embedded in the leader's feasible set. Second, the partition-refinement signal is therefore not only a dual vector of a recourse function, but information about the follower's optimal basis and the resulting leader–follower interaction. This motivates a bilevel-aware refinement mechanism that separates scenarios according to structural changes in the follower's response.

3. The Adaptive Partitioning Algorithm

We begin this section by describing the Adaptive Partitioning Method (APM) in its original version for two-stage stochastic programs with recourse. Such a problem has the structure

$$\min_{x\in\mathcal{X}} \quad c^\top x + \mathbb{E}\left[Q(x;\xi)\right], \tag{1}$$

where $\mathcal{X} \subseteq \mathbb{R}^n$ is a nonempty bounded polyhedron, ξ is a random vector with support in Ξ (assumed finite but arbitrarily large), and

$$\begin{aligned} Q(x,\xi) := \min_{y\in\mathbb{R}^m} \quad & d^\top y \\ \text{s.t.} \quad & By = b_\xi - T_\xi x, \\ & y \geq 0, \end{aligned}$$

is the second-stage problem. The variable $y := y(x,\xi)$ is known as the recourse variable associated with first-stage

decision x and scenario ξ. The model captures a sequential decision process: the first-stage decision x must be taken prior to observing the realisation of ξ, and the recourse decision y adapts once ξ is observed. In the literature, x is often referred to as a *here-and-now* decision and y as a *wait-and-see* decision.

Throughout this work, we adopt a discrete distribution over a finite scenario set. If $N = |\Xi|$, an equivalent reformulation of (1) is

$$\begin{aligned} \min_{x\in\mathcal{X}} \quad & c^\top x + \frac{1}{N}\sum_{i=1}^{N} d^\top y_i \\ \text{s.t.} \quad & T_i x + B y_i = b_i, \quad i = 1,\dots,N, \\ & y_i \geq 0, \quad i = 1,\dots,N, \end{aligned} \tag{2}$$

where T_i and b_i denote the realisations of T and b for scenario $\xi_i \in \Xi$, and y_i is the corresponding second-stage solution. Formulation (2) is commonly referred to as the *extended formulation*. When N is moderate, there exist many practical solution methods for (2) (e.g., Benders decomposition, progressive hedging, L-shaped methods). However, when N becomes very large these approaches may become computationally expensive. The APM builds on the observation that many scenarios are structurally similar and may be treated in aggregated form rather than individually, leading to an iterative method that solves master problems akin to (2) but of considerably smaller size.

To develop the APM we first recall the notion of a partition.

Definition 3.1. A partition $\mathcal{A}$ of a set A is a collection of subsets of A, that is $\mathcal{A} = \{A_1, A_2, \dots\}$, satisfying that for every $a \in A$ there is exactly one $i \in \{1, 2, \dots\}$ such that $a \in A_i$.

Consider now a partition $\mathcal{P}$ of the scenario set Ξ. By the law of total expectation, problem (1) can be written as

$$\min_{x\in\mathcal{X}} \quad c^\top x + \sum_{P\in\mathcal{P}} \mathbb{E}\left[Q(x;\xi) \mid P\right]\mathbb{P}(P). \tag{3}$$

For $P \in \mathcal{P}$ define $b_P := \mathbb{E}[b_\xi \mid P]$ and $T_P := \mathbb{E}[T_\xi \mid P]$. The aggregated second-stage subproblem associated with P is

$$\begin{aligned} Q(x,P) := \min_{y\in\mathbb{R}^m} \quad & d^\top y \\ \text{s.t.} \quad & By = b_P - T_P x, \\ & y \geq 0. \end{aligned}$$

The corresponding master problem for partition $\mathcal{P}$ is

$$\min_{x\in\mathcal{X}} \quad c^\top x + \sum_{P\in\mathcal{P}} Q(x,P)\mathbb{P}(P). \tag{4}$$

The APM aims to identify a partition $\mathcal{P}$ such that an optimal solution of (4) is also optimal for (3).

3.1. APM for two-stage stochastic programs

The APM is an iterative algorithm in which, at each iteration, the master problem (4) is solved for the current partition and the partition is refined if necessary.

Definition 3.2. Given a partition $\mathcal{P}$, a refinement of $\mathcal{P}$ is another partition $\mathcal{P}'$ such that, for every $R \in \mathcal{P}'$, there exists $P \in \mathcal{P}$ with $R \subseteq P$.

For the APM it is convenient to assume that every second-stage subproblem $Q(x;\xi)$ has finite optimal value. In that case, strong duality holds and, for any $\xi \in \Xi$, we may work with the dual

$$\begin{aligned} Q_D(x,\xi) := \max_{\lambda\in\mathbb{R}^\ell} \quad & (b_\xi - T_\xi x)^\top \lambda \\ \text{s.t.} \quad & B^\top \lambda \leq d, \end{aligned}$$

and similarly the dual of the aggregated subproblem,

$$\begin{aligned} Q_D(x,P) := \max_{\lambda\in\mathbb{R}^\ell} \quad & (b_P - T_P x)^\top \lambda \\ \text{s.t.} \quad & B^\top \lambda \leq d. \end{aligned}$$

The feasible region of the dual does not depend on the first-stage variable x nor on the scenario realisation. From this it follows that $Q(x;\xi)$ and $Q(x;P)$ are convex in x and therefore Jensen's inequality implies

$$Q(x;\mathbb{E}[\xi]) \leq \mathbb{E}\left[Q(x;\xi)\right],$$

and the same inequality holds conditional on any subset P. Consequently, for any partition $\mathcal{P}$, the optimal value of (4) defines a valid lower bound for (3). Moreover, if $\bar{x}$ is optimal for (4), then $\bar{x}$ is feasible for (3), and evaluating it there produces an upper bound.

Building on this idea, Ramirez-Pico and Moreno (2022) show that if $\bar{x}$ is a fixed first-stage vector, λ_ξ is an optimal dual solution for each scenario, and $\mathcal{P}$ is a partition such that, for all $P \in \mathcal{P}$,

$$\begin{aligned} \mathbb{E}\left[b_\xi^\top \lambda_\xi \mid P\right] &= b_P^\top \mathbb{E}\left[\lambda_\xi \mid P\right], \\ \mathbb{E}\left[(T_\xi \bar{x})^\top \lambda_\xi \mid P\right] &= (T_P \bar{x})^\top \mathbb{E}\left[\lambda_\xi \mid P\right], \end{aligned} \tag{5}$$

then $\mathbb{E}\left[Q(\bar{x};\xi) \mid P\right] = Q(\bar{x};P)$.

This condition yields a natural refinement rule: group scenarios whose optimal second-stage dual solutions are identical. That is, two scenarios ξ_1 and ξ_2 are aggregated into the same element P if $\lambda_{\xi_1} = \lambda_{\xi_2}$. With this grouping condition, (5) holds by construction. The result implies that if $\bar{x}$ is optimal for the master (4) and (5) holds for every $P \in \mathcal{P}$, then $\bar{x}$ is optimal for the original problem (1).

The APM can be stated as follows.

In practice, the refinement step can be implemented by processing each element $P \in \mathcal{P}$ independently, computing dual solutions for each scenario and grouping identical duals, as shown below.

The inner loop induces a partition of each $P \in \mathcal{P}$. If the refinement is effective then $|\mathcal{P}'| > |\mathcal{P}|$. Since $|\Xi|$ is finite, the APM must terminate in a finite number of iterations. In the worst case, the process refines until each scenario is a singleton set, recovering the extended formulation (2). If at

Algorithm 1: APM for two-stage stochastic programs

```
Initialize P = {Ξ}, U = +∞, L = −∞;
while L < U do
    Solve master problem (4) for partition P, obtaining x̄;
    Update L with the optimal value of (4), if appropriate;
    Refine P producing P′;
    Update U := c^⊤ x̄ + E[Q(x̄; ξ)], if appropriate;
    Update P := P′;
end
```

Algorithm 2: Refinement process by dual signatures

```
Initialize partition P and fixed first-stage vector x̄;
for P ∈ P do
    for ξ ∈ P do
        Solve Q_D(x̄; ξ) with solution λ_ξ;
        Add ξ to subset P̄ ⊆ P such that for each ξ̄ ∈ P̄ we have λ_ξ = λ_ξ̄;
        If no such P̄ exists create a new subset;
    end
end
```

some iteration the refinement is not effective, i.e., $\mathcal{P}' = \mathcal{P}$, then Ramirez-Pico and Moreno (2022) prove that $L = U$ and the algorithm has reached optimality for the original problem. The convexity of $Q(x;\xi)$ is fundamental in these arguments, as it permits the construction of lower bounds and the convergence proof.

3.2. Stochastic bilevel problems

A bilevel problem can also be interpreted in the form of a two-stage program, because two types of variables are determined sequentially. The main difference is that in bilevel models the relevant second-stage information is the optimal decision of the follower, not only its optimal value.

Analogous to (1), consider the stochastic bilevel model

$$\min_{x\in\mathcal{X}} \quad c^\top x + \mathbb{E}\left[Q(x;\xi)\right], \tag{6}$$

where again $\mathcal{X} \subseteq \mathbb{R}^n$ is a nonempty bounded polyhedron and ξ is a random vector with finite support. The lower level is now described by

$$\begin{aligned} Q(x,\xi) := \min_{y\in\mathbb{R}^m} \quad & d^\top y \\ \text{s.t.} \quad & y \in \arg\min_{z\geq 0} \left\{ g_{x,\xi}^\top z : \; Bz \leq T_\xi x + f_\xi \right\}. \end{aligned}$$

The inner argmin indicates that the follower may have multiple optimal solutions. Common modelling choices include the optimistic case (the follower selects the leader-favourable optimal response) and the pessimistic case (the follower selects the least favourable one). In this work we assume that, for all $x \in \mathcal{X}$, the follower problem has a nonempty feasible set and, with probability one, a unique optimal solution. That is,

$$\mathbb{P}\left(\xi \in \Xi : \left|\arg\min_{z\geq 0} \left\{ g_{x,\xi}^\top z : \; Bz \leq T_\xi x + f_\xi \right\}\right| = 1\right) = 1.$$

When Ξ is finite the probabilistic statement reduces to a statement that holds for all scenarios.

A common reformulation of $Q(x;\xi)$ exploits complementarity slackness for the follower problem, yielding

$$\begin{aligned} Q(x,\xi) := \min_{y\in\mathbb{R}^m} \quad & d^\top y \\ \text{s.t.} \quad & 0 \leq T_\xi x + f_\xi - By \perp \pi \geq 0, \\ & 0 \leq g_{\xi,x} + B^\top \pi \perp y \geq 0. \end{aligned}$$

Applying the law of total expectation over a partition $\mathcal{P}$ yields

$$\min_{x\in\mathcal{X}} \quad c^\top x + \sum_{P\in\mathcal{P}} \mathbb{E}\left[Q(x;\xi) \mid P\right] \mathbb{P}(P). \tag{7}$$

Assuming again a finite uniform scenario set $\Xi = \{\xi_1, \ldots, \xi_N\}$, the equivalent one-level extended formulation becomes

$$\begin{aligned} \min_{x\in\mathcal{X}} \quad & c^\top x + \sum_{P\in\mathcal{P}} \sum_{i\in P} \frac{|P|}{N} d^\top y_i \\ \text{s.t.} \quad & 0 \leq T_i x + f_i - By_i \perp \pi_i \geq 0, \quad i = 1, \ldots, N, \\ & 0 \leq g_{i,x} + B^\top \pi_i \perp y_i \geq 0, \quad i = 1, \ldots, N, \end{aligned} \tag{8}$$

where $T_i := T_{\xi_i}$, $f_i := f_{\xi_i}$ and $g_{i,x} := g_{\xi_i,x}$. For a cell P of the partition, define the aggregated bilevel subproblem

$$\begin{aligned} Q(x,P) := \min_{y\in\mathbb{R}^m} \quad & d^\top y \\ \text{s.t.} \quad & 0 \leq T_P x + f_P - By \perp \pi \geq 0, \\ & 0 \leq g_{P,x} + B^\top \pi \perp y \geq 0, \end{aligned}$$

where $T_P := \mathbb{E}[T_\xi \mid P]$, $f_P := \mathbb{E}[f_\xi \mid P]$ and $g_{P,x} := \mathbb{E}[g_{\xi,x} \mid P]$. The master problem for stochastic bilevel programs is then

$$\min_{x\in\mathcal{X}} \quad c^\top x + \sum_{P\in\mathcal{P}} Q(x;P)\mathbb{P}(P). \tag{9}$$

We are interested in conditions on $\mathcal{P}$ ensuring $\mathbb{E}[Q(x;\xi) \mid P] = Q(x;P)$ for every cell $P \in \mathcal{P}$.

Proposition 3.3. *Let $\bar{x}$ be feasible in (9) and let $\{y_\xi\}_{\xi\in\Xi}$ and $\{\pi_\xi\}_{\xi\in\Xi}$ be the primal and dual solutions of $Q(\bar{x};\xi)$ for each $\xi \in \Xi$. Assume the solution set is singleton for every scenario. If, for a cell P, the equalities*

$$\begin{aligned} &\mathbb{E}\left[(T_\xi \bar{x} + f_\xi - By_\xi)^\top \pi_\xi \mid P\right] \\ &\quad = \mathbb{E}\left[T_\xi \bar{x} + f_\xi - By_\xi \mid P\right]^\top \mathbb{E}\left[\pi_\xi \mid P\right] \end{aligned} \tag{10}$$

and

$$\begin{aligned}&\mathbb{E}\left[(g_{\xi,\bar{x}} + B^\top \pi_\xi)^\top y_\xi \mid P\right] \\ &\quad = \mathbb{E}\left[g_{\xi,\bar{x}} + B^\top \pi_\xi \mid P\right]^\top \mathbb{E}\left[y_\xi \mid P\right]\end{aligned} \tag{11}$$

hold, then

$$Q(\bar{x}; P) \le \mathbb{E}\left[Q(\bar{x}; \xi) \mid P\right].$$

Moreover, if for each $P \in \mathcal{P}$ the primal solution $\bar{y}_P$ of $Q(\bar{x}; P)$ is unique, then equality holds.

Proof. Since y_ξ and π_ξ are feasible for the problem defining $Q(\bar{x}; \xi)$, the complementarity relations imply

$$(T_\xi \bar{x} + f_\xi - B y_\xi)^\top \pi_\xi = 0$$

and

$$(g_{\xi,\bar{x}} + B^\top \pi_\xi)^\top y_\xi = 0.$$

Let $y_P := \mathbb{E}[y_\xi \mid P]$ and $\pi_P := \mathbb{E}[\pi_\xi \mid P]$. Then

$$\begin{aligned}&(T_P \bar{x} + f_P - B y_P)^\top \pi_P \\ &\quad = (\mathbb{E}[T_\xi \mid P]\bar{x} + \mathbb{E}[f_\xi \mid P] - B\mathbb{E}[y_\xi \mid P])^\top \\ &\qquad \mathbb{E}[\pi_\xi \mid P] \\ &\quad = \mathbb{E}[T_\xi \bar{x} + f_\xi - B y_\xi \mid P]^\top \\ &\qquad \mathbb{E}[\pi_\xi \mid P] \\ &\quad = \mathbb{E}\Big[(T_\xi \bar{x} + f_\xi - B y_\xi)^\top \\ &\qquad \pi_\xi \mid P\Big] \\ &\quad = 0.\end{aligned}$$

Similarly,

$$\begin{aligned}&(g_{P,\bar{x}} + B^\top \pi_P)^\top y_P \\ &\quad = (\mathbb{E}[g_{\xi,\bar{x}} \mid P] + B^\top \mathbb{E}[\pi_\xi \mid P])^\top \\ &\qquad \mathbb{E}[y_\xi \mid P] \\ &\quad = \mathbb{E}[g_{\xi,\bar{x}} + B^\top \pi_\xi \mid P]^\top \\ &\qquad \mathbb{E}[y_\xi \mid P] \\ &\quad = \mathbb{E}\Big[(g_{\xi,\bar{x}} + B^\top \pi_\xi)^\top \\ &\qquad y_\xi \mid P\Big] \\ &\quad = 0.\end{aligned}$$

Therefore, y_P and π_P are feasible for the problem defining $Q(\bar{x}; P)$. Hence,

$$Q(\bar{x}; P) \le d^\top y_P.$$

Moreover,

$$\begin{aligned}d^\top y_P &= d^\top \mathbb{E}[y_\xi \mid P] \\ &= \mathbb{E}[d^\top y_\xi \mid P] \\ &= \mathbb{E}[Q(\bar{x}; \xi) \mid P].\end{aligned}$$

Algorithm 3: APM for stochastic bilevel programs

```
Initialize P = {Ξ} and set U = +∞;
while conditions (10) and (11) do not hold do
    Solve master problem (9) for partition P,
      obtaining x̄;
    Refine P to obtain P′ based on primal or dual
      signatures as dictated by Assumption 3.4;
    Update U := c^⊤ x̄ + E[Q(x̄; ξ)], if appropriate;
    Update P := P′;
end
```

Consequently,

$$Q(\bar{x}; P) \le \mathbb{E}[Q(\bar{x}; \xi) \mid P].$$

Finally, if the primal solution of $Q(\bar{x}; P)$ is unique, then $\bar{y}_P = y_P$. Therefore,

$$Q(\bar{x}; P) = d^\top \bar{y}_P = d^\top y_P,$$

which proves the equality statement. □

The main difference with the two-stage setting is that here special conditions on $\mathcal{P}$ are required to obtain lower bounds, since $Q(x; \xi)$ is not necessarily convex. Proposition 3.3 provides guidance for constructing valid lower bounds for the original bilevel problem.

From now on we adopt the following assumption.

Assumption 3.4. The uncertainty affects either the right-hand side of the follower problem or its objective but not both. In other words, either

1. $T_\xi \equiv T$ and $f_\xi \equiv f$, or
2. $g_{\xi,x} \equiv g_x$ for all ξ.

This assumption ensures that one side of the follower feasible region, primal or dual, remains fixed across scenarios. When the primal region is fixed and the partition is constructed so that y_ξ is constant over each cell P, conditions (10) and (11) hold. Similarly, if the dual region is fixed and the partition groups scenarios with identical dual π_ξ, the conditions also hold. These observations motivate the following variant of the APM for stochastic bilevel programs.

The refinement step may be executed by grouping scenarios with identical primal bases or identical dual solutions, depending on which region is fixed. In the next subsection we present an instructive example showing why a straightforward application of the APM may stall and how basis-aware refinement overcomes this issue.

3.3. Continuous bilevel knapsack example

Consider the continuous bilevel knapsack model

$$\begin{aligned}&\max_{x \in [x^-, x^+]} \; d^\top y \; - \; c\, x, \\ &\quad \text{s.t.} \quad y \in \arg\max\{g^\top y : a^\top y \le x,\; y \in [0,1]^n\},\end{aligned} \tag{12}$$

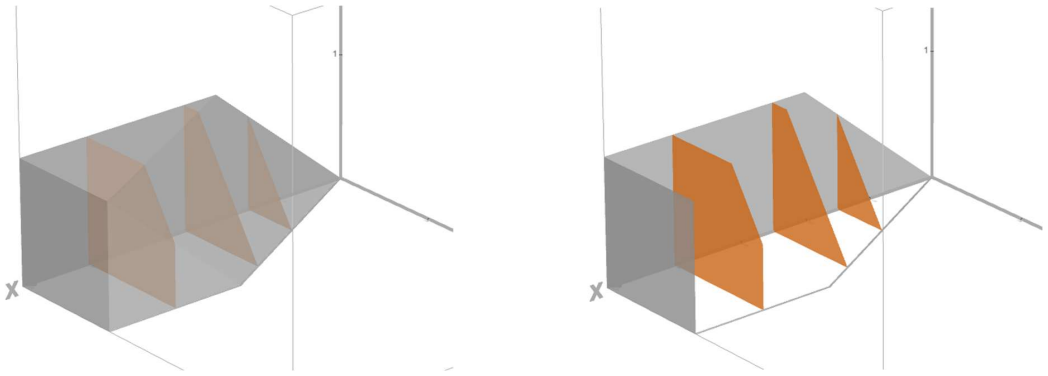

Figure 1: Illustrative geometry of the continuous bilevel knapsack for $a = (3, 2)$.

where the leader chooses the knapsack capacity x and the follower chooses fractional item quantities y to maximise its own profit. The follower choices influence the leader objective because the leader values items differently. In the stochastic setting, the follower profit vector g is random and continuous, so that with probability one the follower optimum is unique. In our finite-sample approach we assume an arbitrarily large sample of realisations.

Given x, the follower feasible set is $Y(x) = \{y \mid a^\top y \le x,\ y \in [0,1]^n\}$ and the shared feasible region is $\Omega = \{(x, y) \mid a^\top y \le x,\ y \in [0,1]^n\}$. Figure 1 illustrates these sets for the toy example $a = (3, 2)$. As we now show, the degenerate vertices that appear at certain break points of the leader objective can make a direct application of the APM stall.

To see how a direct application of the APM can fail, note that the follower's reaction depends only on the sign structure of its profit vector: if, over a cell P, the aggregated profit vector $g_P = \mathbb{E}[g_\xi \mid P]$ has non-positive components, the reaction set $R(x) = \arg\max\{g_\xi^\top y : y \in Y(x)\}$ collapses to the origin, and the leader, facing a null follower response, selects the cheapest capacity $x = x^-$.

Example 3.5. Let $a = (3, 2)$, $c = 3$, $d = (15, 15)$ and $[x^-, x^+] = [0, 5]$, with three equiprobable scenarios whose follower profit vectors are $g^1 = (-1, -1)$, $g^2 = (-1, 1)$ and $g^3 = (1, -1)$. Starting the APM from the trivial partition $\mathcal{P} = \{\Xi\}$, the aggregated profit vector is $g_\Xi = \frac{1}{3}(g^1 + g^2 + g^3) = (-\frac{1}{3}, -\frac{1}{3})$, both of whose components are negative. The aggregated follower response is therefore $y = 0$ for every x, and the master problem returns $\bar{x} = 0$. When refining at $\bar{x} = 0$, the capacity constraint $a^\top y \le 0$ forces $y_\xi = 0$ in every scenario, so all scenarios share the same primal follower solution and a primal-based refinement leaves the partition unchanged, $\mathcal{P}' = \mathcal{P}$. The APM thus stalls at $x = 0$, even though the true optimum is $x^* = 2$ with expected leader value $\frac{7}{3}$.

The reason for the stall is geometric. At $\bar{x} = 0$ the follower's cut $Y(0) = \{0\}$ reduces to a single *degenerate* vertex: in $\mathbb{R}^2$ the point $(0, 0)$ is the intersection of the three hyperplanes $y_1 = 0$, $y_2 = 0$ and $a^\top y = 0$, that is, of $n + 1$ hyperplanes rather than n. At such a vertex the primal solution is common to all scenarios, but the *optimal basis* is not: it records which constraints the follower holds active, and hence the direction in which it would move if the capacity were relaxed. Writing the follower problem with a slack $s \ge 0$ for the capacity constraint $a^\top y + s = x$ and slacks $z_k \ge 0$ for the bounds $y_k + z_k = 1$, the optimal bases at $\bar{x} = 0$ are $\mathcal{B}_1 = \{s, z_1, z_2\}$, $\mathcal{B}_2 = \{y_2, z_1, z_2\}$ and $\mathcal{B}_3 = \{y_1, z_1, z_2\}$, one per scenario. A refinement that separates scenarios by their optimal basis therefore splits Ξ into three singletons, and the next master problem coincides with the extended formulation (8), recovering the global optimum $x^* = 2$.

This motivates replacing the primal-based refinement by a basis-aware rule that groups scenarios according to their optimal bases rather than their primal values, thereby isolating the structural regions relevant to the leader instead of stalling at a degenerate capacity. The procedure is given below.

Algorithm 4: Refinement by optimal bases for stochastic bilevel programs

```
Initialize partition 𝒫 and fixed x̄;
for P ∈ 𝒫 do
    for ξ ∈ P do
        Solve Q(x̄; ξ) and obtain optimal basis ℬ_ξ;
        Add ξ to subset P̄ ⊆ P such that for each
          ξ̄ ∈ P̄ we have ℬ_ξ = ℬ_ξ̄;
        If no such P̄ exists create a new subset;
    end
end
```

This refinement ensures that scenarios inducing the same optimal basis are grouped while scenarios inducing different bases are separated, so that only structurally distinct follower responses are represented individually.

This behaviour has a clean reading in terms of the chamber complex of Muñoz et al. (2025). The leader objective $c^\top x + \mathbb{E}[Q(x;\xi)]$ is piecewise linear, with break points at the capacities where the follower subproblem has a degenerate vertex. For the instance of Example 3.5 these are $x \in \{0, 2, 3, 5\}$, giving the chamber complex $\{[0, 2], [2, 3], [3, 5]\}$, and the optimum $x^* = 2$ lies on a chamber boundary. Solving the master problem for a fixed partition amounts to exploring one chamber, and basis-aware refinement is what carries the algorithm across chamber boundaries toward the break points holding the leader's candidate optima.

3.4. Starting bias

The first-stage decision in the APM comes from solving the master problem for the trivial partition $\mathcal{P} = \{\Xi\}$. Subsequent refinements proceed from this initial choice and may introduce bias or stalling. To mitigate this effect, we propose three modifications, described below. The first alters how the partition evolves across iterations, while the remaining two replace the trivial starting partition with a non-trivial one built before the first master solve. All three are evaluated in Section 4.

Scenario bundling (refinement from scratch) Rather than nesting successive refinements within the current cells, this variant discards the partition after each master solution and rebuilds it from scratch from the follower responses evaluated at the incumbent first-stage decision. Because every cell is reformed at each iteration, groups that an earlier refinement had split unnecessarily can be merged again, so the partition is not forced to grow monotonically. This keeps the number of scenario blocks compact and the master problems small, at the cost of recomputing the grouping in full at every iteration.

K-means initial partitions This variant seeds the algorithm with a non-trivial partition obtained by clustering the scenario data before any master problem is solved. Concretely, the scenario cost vectors $\{g_e\}_{e=1}^{p}$ are grouped with K-means into $K_0 = \lfloor 1.8\sqrt{p} \rfloor$ initial blocks, so that scenarios with similar follower objectives start in the same cell. The intent is to expose heterogeneous follower responses from the outset—information that a single trivial block $\{\Xi\}$ averages away—thereby reducing the number of refinement iterations needed to reach an accurate partition.

Central initialization Instead of clustering the raw data, this variant builds the initial partition from a representative follower response. The first-stage vector is fixed at a central value of its feasible range—here the midpoint $x_0 = \frac{1}{2}\sum_j a_j$ of the leader's capacity interval—and the follower problem is solved at x_0 for every scenario; the resulting follower structure (primal solutions or optimal bases) is then used to form the starting blocks. This provides a structurally informed partition at a single evaluation point, typically yielding fewer initial blocks than K-means while still avoiding the bias of the trivial partition.

Section 4 details these modifications and reports computational results comparing their performance.

4. Computational results

This section evaluates the computational performance of the adaptive partitioning algorithm described in the previous section. The experiments assess whether scenario aggregation and refinement can approximate the deterministic equivalent formulation while reducing the computational effort required to solve stochastic bilevel linear programs. The extended formulation is used as the reference benchmark in all comparisons and is solved with the same solver settings as the aggregated models.

The section is organized as follows. Section 4.1 describes the test bed, implementation details, algorithmic variants, and performance metrics. Section 4.2 compares the solution quality of all variants against the extended formulation. Section 4.3 studies robustness under objective-value tolerances and reports the full distribution of objective gaps. Section 4.4 reports running times, speed-ups, iteration counts, and final partition sizes. Section 4.5 discusses the algorithmic mechanisms behind the observed performance patterns through paired class-level averages, and Section 4.6 summarizes the main conclusions.

4.1. Experimental design

The computational study is based on stochastic bilevel continuous knapsack instances. We consider three instance classes with increasing size: 4 items and 1,600 scenarios, 6 items and 2,400 scenarios, and 8 items and 3,200 scenarios. For each class, 200 independent random seeds are generated, and every algorithmic variant is evaluated on the same set of seeds. The resulting adaptive-partitioning data set contains 3,600 runs: six algorithms, three instance classes, and 200 replications per algorithm–class pair. This balanced design separates the effect of the partitioning strategy from random instance variability.

The instance-generation protocol is fixed across all algorithms. In each replication, the deterministic knapsack parameters are generated first. The leader has a single decision variable x, representing the capacity purchased for the follower's knapsack. The capacity cost is sampled as $c \sim \mathcal{U}\{1, \dots, 19\}$. Item weights are sampled independently as $a_j \sim \mathcal{U}\{1, \dots, 19\}$, and the leader valuation of item j is constructed as $d_j = a_j \rho_j$, where $\rho_j \sim \mathcal{U}\{15, \dots, 39\}$. Thus, more resource-consuming items tend to have larger leader-side valuations, while still preserving heterogeneity across items.

Uncertainty enters only through the follower's objective coefficients. For each scenario $e = 1, \dots, p$, the cost vector $g_e = (g_{1,e}, \dots, g_{m,e})$ is sampled independently from the continuous uniform distribution $g_{j,e} \sim \mathcal{U}[-100, 100]$. All scenarios are assigned probability $1/p$. The lower-level feasible set is represented by $B = [a^\top; I_m]$, $f = (0, 1, \dots, 1)^\top$, and $T = (1, 0, \dots, 0)^\top$, so that $By_e \le Tx + f$ encodes both the knapsack capacity constraint and the upper bound $y_{j,e} \le 1$. The leader capacity is bounded by $0 \le x \le \sum_{j=1}^{m} a_j$.

Each adaptive-partitioning run is compared with the extended-formulation benchmark recorded for the corresponding seed and instance class. Standard variants start from the trivial partition containing all scenarios in one block. The K-Means initialization instead clusters the scenario vectors g_e into $K_0 = \lfloor 1.8\sqrt{p} \rfloor$ initial blocks, whereas the central initialization first evaluates the follower response at the midpoint of the leader's capacity range $0 \le x \le \sum_{j=1}^{m} a_j$, namely $x_0 = \frac{1}{2}\sum_{j=1}^{m} a_j$, and uses the resulting structure to initialize the partition. In all formulations, the Big-M constant is set to 10^6, and the norm-based grouping criterion uses tolerance 10^{-5} when comparing follower solution vectors.

All experiments were run on a workstation equipped with an Intel Core i9-14900KF processor, 128 GB of DDR5-5200 RAM, and an NVMe PCIe 4.0 SSD. The algorithms were implemented in Python 3.10.16, and all optimization models were solved with IBM ILOG CPLEX Studio 22.1.1 through its Python interface. Wall-clock times are reported in seconds. Code, instance-generation scripts, and numerical results are available at https://github.com/cmunozrey/adaptive-scenario-partitioning-sblp.

The experiments compare six variants. The standard adaptive partitioning method starts from the trivial partition and refines blocks using the norm-based similarity of follower solutions. The basis-aware variant replaces this criterion with the optimal basis status of the lower-level variables, so that scenarios are split only when they induce structurally different follower bases. The two scenario-bundling variants rebuild the partition from scratch after each master solution, either using the standard criterion or the basis-aware criterion. The last two variants combine basis-aware refinement with a non-trivial initial partition: one obtained by K-Means clustering of scenario cost vectors and one obtained by evaluating the follower response at the center of the leader's capacity range (half of the total item weight). This organization keeps the comparison focused: the extended formulation provides a common benchmark, the six adaptive variants differ only in the partition-management mechanism, and all data-generation, numerical, and solver settings are held fixed across algorithms.

Let x^A and z^A denote the leader decision and objective value returned by an adaptive partitioning variant, and let x^E and z^E be the corresponding values obtained from the extended formulation. We report the leader-decision gap Δ_x and the objective-value gap Δ_z, defined as

$$\Delta_x = 100\,\frac{|x^A - x^E|}{\max\{1, |x^E|\}}, \qquad \Delta_z = 100\,\frac{|z^A - z^E|}{\max\{1, |z^E|\}}.$$

We also report the total running time, the master-problem time, the refinement time, the number of iterations, and the final number of scenario blocks. Two success indicators are used: exact agreement with the extended formulation, which requires both the leader decision and the objective value to match the reference solution, and objective-value agreement within prescribed tolerances.

4.2. Accuracy against the extended formulation

Table 1 summarizes the main accuracy results. The standard method recovers the extended-formulation solution in a substantial fraction of the runs, but its performance is sensitive to the trivial initial partition and to numerical differences in the induced follower solutions. In the 4-item class, the exact agreement rate is 55.5%, the mean leader-decision gap is 14.064%, and the mean objective-value gap is 10.428%. The objective gap decreases in the larger test classes, reaching 0.458% in the 8-item class, whereas exact agreement with the extended formulation becomes less frequent as the instances grow. This distinction indicates that matching the exact leader capacity is a stricter requirement than matching the reference objective value.

Basis-aware refinement substantially improves stability. For the 4-item class, exact agreement increases from 55.5% to 74.0%, while the mean objective-value gap decreases from 10.428% to 0.320%. The same pattern is observed for the two larger classes: objective gaps fall from 2.739% to 0.078% in the 6-item class and from 0.458% to 0.044% in the 8-item class. This supports the computational value of grouping scenarios according to the structure of the follower's optimal basis rather than according to small differences in continuous solution vectors. The basis signal provides a more structural refinement criterion for the lower-level response and is therefore less sensitive to degeneracy and numerical perturbations in the lower-level subproblems.

The most accurate variants are those that combine a non-trivial initial partition with basis-aware refinement. K-Means with basis-aware refinement achieves the highest exact agreement in every instance class, with rates of 84.0%, 69.5%, and 57.5%. It also maintains very small mean objective gaps, between 0.021% and 0.036%. The central initialization is slightly less accurate in the leader decision but similarly strong in objective value, with gaps between 0.033% and 0.051%. These results indicate that providing the master problem with a structurally informative initial partition is valuable in stochastic bilevel instances, where a single initial block can mask heterogeneous follower responses.

Scenario bundling has a different role. It rebuilds scenario bundles after each master solution and produces much smaller final partitions, but this speed-oriented mechanism weakens exact agreement when it is applied to the standard norm-based refinement. In the 8-item class, exact agreement decreases from 44.5% to 34.5% when bundling is added to the standard method. When bundling is combined with basis-aware refinement, the loss is much smaller: the exact agreement rate changes from 50.5% to 50.0%, and the mean objective gap remains close to 0.05%. Thus, bundling is best interpreted as a computational-efficiency device whose accuracy depends on whether the refinement criterion already captures the structural response of the follower.

4.3. Robustness under objective-value tolerances

Exact equality with the extended formulation is a stringent criterion. In stochastic bilevel instances, two leader capacities can induce nearly indistinguishable objective values

Table 1
Accuracy with respect to the extended formulation. Exact denotes the percentage of runs in which both the leader decision and the objective value match the extended-formulation benchmark. Mean gaps are reported in percent.

	4 items, 1,600 scenarios			6 items, 2,400 scenarios			8 items, 3,200 scenarios		
Algorithm	Exact	x gap	Obj. gap	Exact	x gap	Obj. gap	Exact	x gap	Obj. gap
Standard APM	55.5	14.064	10.428	52.5	5.179	2.739	44.5	2.539	0.458
Basis-aware APM	74.0	2.611	0.320	60.5	1.804	0.078	50.5	1.688	0.044
Standard APM + SB	51.5	14.878	10.486	48.0	5.714	2.797	34.5	3.095	0.504
Basis-aware APM + SB	72.5	2.904	0.325	56.5	1.865	0.077	50.0	1.790	0.050
K-Means + BA	84.0	0.832	0.021	69.5	1.157	0.028	57.5	1.458	0.036
Central + BA	78.0	1.485	0.051	63.0	1.348	0.033	54.0	1.560	0.037

Table 2
Percentage of runs with objective-value gap within a prescribed tolerance of the extended formulation.

	4 items, 1,600 scenarios			6 items, 2,400 scenarios			8 items, 3,200 scenarios		
Algorithm	0.1%	1%	10%	0.1%	1%	10%	0.1%	1%	10%
Standard APM	67.0	82.0	87.5	74.0	92.5	97.5	80.0	98.5	99.5
Basis-aware APM	85.5	95.5	99.0	84.0	99.0	100.0	85.0	100.0	100.0
Standard APM + SB	62.5	80.0	87.5	68.5	92.0	97.5	73.0	97.5	99.5
Basis-aware APM + SB	84.5	95.0	99.0	81.5	99.0	100.0	83.5	99.5	100.0
K-Means + BA	92.5	100.0	100.0	89.5	100.0	100.0	88.0	100.0	100.0
Central + BA	90.0	99.5	100.0	89.0	100.0	100.0	88.0	100.0	100.0

even when the capacity itself differs. This is particularly relevant when the lower-level problem has multiple optimal or nearly optimal bases. Table 2 therefore reports the percentage of runs whose objective-value gap is within 0.1%, 1%, and 10%.

Figure 2 complements Table 2 by showing the full seed-level distribution of objective-value gaps. The vertical axis uses a symmetric logarithmic scale, which displays exact zero gaps together with small and large positive gaps. The boxplots are consequently concentrated at zero for the most accurate variants in the smaller class; this reflects that the central part of the seed distribution matches the extended-formulation objective exactly, while the jittered points show the remaining tail behavior. The horizontal reference lines correspond to the 0.1% and 1% tolerance levels used in the table.

The tolerance-based results show that the proposed variants are much more reliable in objective value than exact agreement alone suggests. At the 0.1% tolerance level, the standard method solves between 67.0% and 80.0% of the instances, whereas the basis-aware method solves between 84.0% and 85.5%. The two informed initial partitions are the strongest alternatives: K-Means with basis-aware refinement solves 92.5%, 89.5%, and 88.0% of the runs within 0.1%, and the central initialization solves 90.0%, 89.0%, and 88.0%. At the 1% tolerance level, all basis-aware variants are highly reliable. K-Means with basis-aware refinement reaches 100% success in all three instance classes, while central initialization with basis-aware refinement reaches at least 99.5%. These results suggest that the proposed partitioning strategies recover the economically relevant value of the leader's solution even when the exact capacity differs from the reference solution.

4.4. Run-time behavior

Table 3 reports the mean total running time of each adaptive variant, the speed-up relative to the corresponding mean extended-formulation time, the mean number of iterations, and the final number of scenario blocks. The speed-up is computed from the benchmark time recorded for the same algorithm–class group, so it is a direct ratio of comparable wall-clock averages. The adaptive partitioning variants provide substantial reductions in wall-clock time relative to the extended formulation. For the class with 4 items and 1,600 scenarios, the extended formulation requires 10.27 seconds on average, while the adaptive variants range from 2.78 to 3.87 seconds. For 6 items and 2,400 scenarios, the extended formulation requires 24.41 seconds, while the adaptive variants range from 6.89 to 11.44 seconds. For 8 items and 3,200 scenarios, the corresponding extended-formulation averages are 85.71 seconds for the standard, basis-aware, and bundling groups and 83.85 seconds for the non-trivial-initialization groups; the adaptive variants range from 18.06 seconds for standard scenario bundling to 50.63 seconds for K-Means with basis-aware refinement. Thus, the adaptive schemes remain faster than solving the full deterministic equivalent even when the refinement process creates several hundred final scenario blocks.

Figure 3 complements the class-level averages in Table 3 by showing the full distribution of wall-clock times across the 200 seeds. The extended formulation is included as a reference method using the benchmark observations indexed by seed and instance class, while the six adaptive-partitioning variants are shown with their corresponding seed-level observations. The vertical axis is reported in seconds on a logarithmic scale, and the dashed horizontal line marks the 900-second time limit imposed on the extended formulation. The figure highlights two features that are hidden by averages alone. First, the adaptive variants

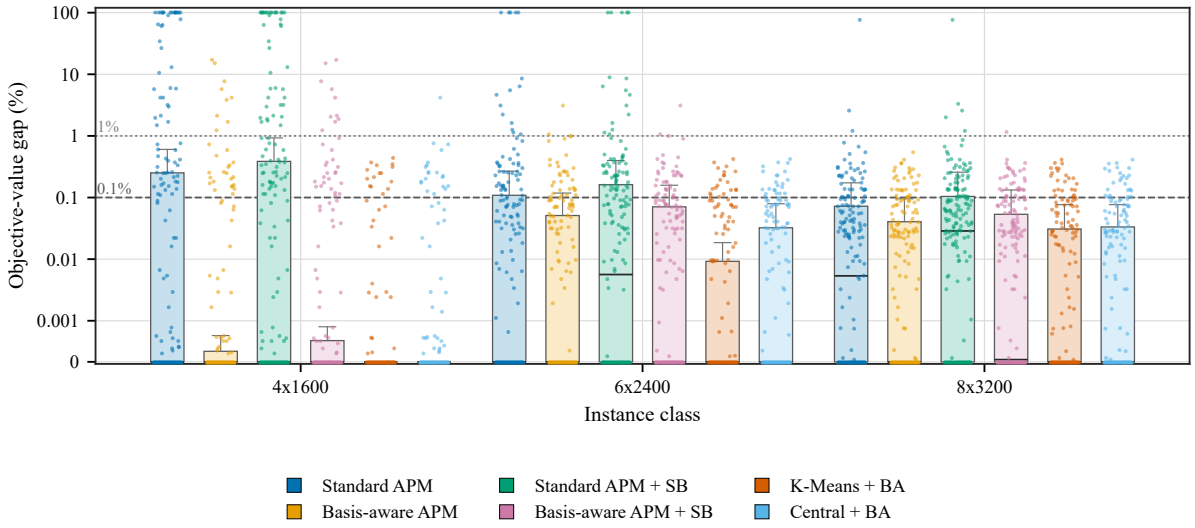

Figure 2: Distribution of objective-value gaps across adaptive-partitioning variants and instance classes. Boxplots and jittered points are computed from the 200 seeds in each method–class group. The vertical axis uses a symmetric logarithmic scale to display exact zero gaps together with positive gaps, and horizontal reference lines indicate 0.1% and 1% objective-gap tolerances.

Table 3
Mean run-time and partition statistics. Time is the mean APM wall-clock time in seconds; Sp. is the ratio between the corresponding mean extended-formulation time and the mean APM time; It. is the mean number of iterations; Blocks is the mean final number of scenario blocks.

	4 items, 1,600 scenarios				6 items, 2,400 scenarios				8 items, 3,200 scenarios			
Algorithm	Time	Sp.	It.	Blocks	Time	Sp.	It.	Blocks	Time	Sp.	It.	Blocks
Standard APM	2.78	3.70	3.45	26.4	7.34	3.33	4.35	188.7	40.25	2.13	4.88	791.4
Basis-aware APM	3.28	3.13	3.79	36.3	8.39	2.91	4.45	214.8	38.87	2.20	4.75	816.3
Standard APM + SB	3.13	3.28	4.00	15.3	7.09	3.44	4.90	75.2	18.06	4.75	5.43	334.4
Basis-aware APM + SB	3.87	2.66	4.55	18.9	8.77	2.78	5.77	83.3	19.34	4.43	5.79	350.6
K-Means + BA	3.23	3.18	2.60	215.5	11.44	2.13	2.84	553.0	50.63	1.66	2.94	1308.0
Central + BA	3.03	3.40	3.45	31.6	6.89	3.54	4.00	172.4	27.52	3.05	4.30	671.8

are concentrated well below the reference limit, whereas the extended formulation exhibits longer right tails, especially in the larger classes. Second, the largest instances display the clearest separation among variants: scenario bundling shifts the runtime distribution downward, while K-Means with basis-aware refinement has a wider and higher distribution because its more informative initial partition creates larger master problems. This pattern is consistent with the role of partition size as the main driver of master-problem complexity.

Figure 4 summarizes the joint accuracy–time trade-off behind Tables 1–3. The figure is computed from the full set of 200 replications in each algorithm–class group, consistently with the averages reorganized later in Table 4. Each panel corresponds to one instance class, and each marker reports the resulting mean objective-value gap and mean speed-up for one algorithm in that class. Moving left indicates a smaller objective-value gap, while moving upward indicates a larger speed-up over the extended formulation. The figure shows that the basis-aware variants dominate the standard norm-based variants in objective-value accuracy: all basis-aware means lie below the 0.5% objective-gap level, and all except Basis-aware APM and Basis-aware APM + SB in the 4-item class lie below 0.1%. It also separates the two main efficiency mechanisms. Scenario bundling provides the highest speed-ups for the largest instances by reducing final partition sizes, whereas the informed initial partitions move the solution toward the low-gap region. Central initialization preserves much of the time advantage, while K-Means is the most accuracy-oriented option.

The results reveal two complementary sources of efficiency. Scenario bundling produces the smallest final partitions among the large instances. In the 8-item class, the standard method ends with 791.4 blocks on average, while standard scenario bundling ends with 334.4 blocks and reduces the mean time from 40.25 to 18.06 seconds. The same effect appears in the basis-aware variants: bundling reduces the final partition from 816.3 to 350.6 blocks and reduces the mean time from 38.87 to 19.34 seconds. Thus, bundling is particularly effective when the cost of repeatedly solving large master problems dominates the total time, although

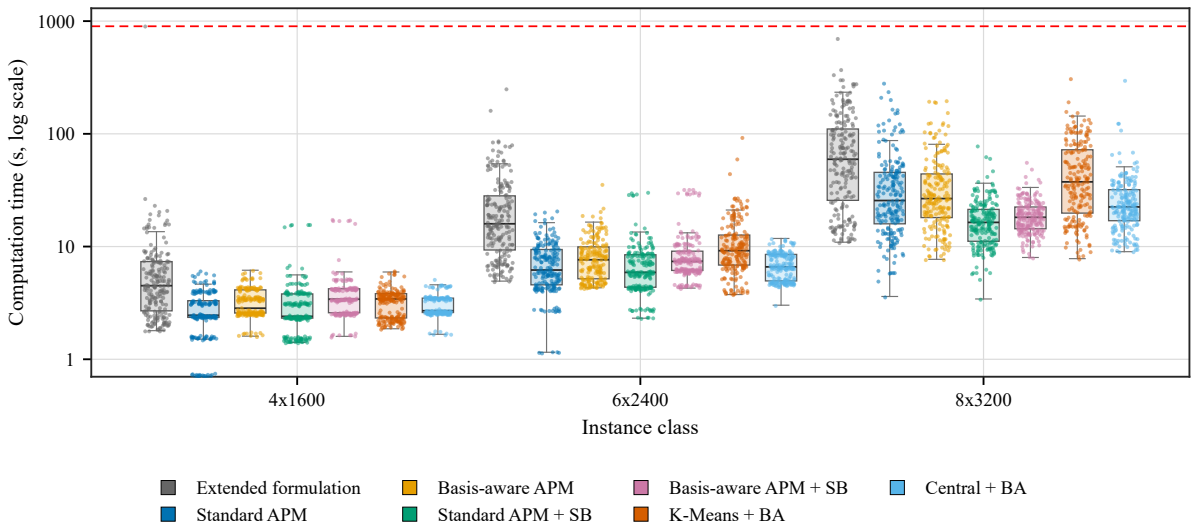

Figure 3: Distribution of computation times across algorithms and instance classes. Boxplots and jittered points are computed from the 200 seeds in each method–class group. The extended formulation is included through the benchmark observations indexed by seed and instance class. The vertical axis reports wall-clock time in seconds on a logarithmic scale, and the dashed horizontal line indicates the 900-second time limit used for the extended formulation.

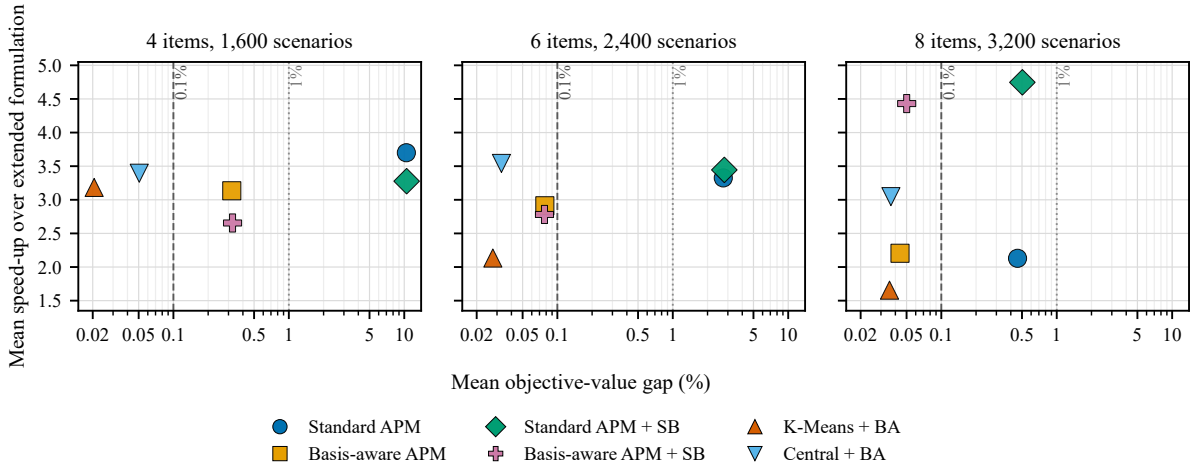

Figure 4: Accuracy–time trade-off across algorithms and instance classes. Each marker is computed from all 200 seeds of the corresponding algorithm–class group. The horizontal axis reports the mean objective-value gap relative to the extended formulation, and the vertical axis reports the speed-up, defined as the ratio of the corresponding mean extended-formulation time to the mean adaptive-partitioning time. Vertical reference lines indicate 0.1% and 1% objective-gap tolerances.

its accuracy is best preserved when paired with a structural refinement criterion.

Informed initial partitions reduce the number of iterations but may increase the size of the early master problems. K-Means with basis-aware refinement has the smallest mean iteration count in all three classes, with 2.60, 2.84, and 2.94 iterations. This accuracy comes at the price of larger master models: the final number of blocks reaches 1,308.0 in the 8-item class, and the mean running time is 50.63 seconds. The central initialization offers a more balanced compromise. It obtains objective gaps comparable to K-Means but requires fewer blocks and less time, especially in the 8-item class, where it averages 27.52 seconds and 671.8 blocks. This suggests that an initialization based on a representative follower response can capture much of the useful structural information without creating as many initial groups as direct clustering of the scenario vectors.

The master/refinement decomposition further clarifies this trade-off. In the 8-item class, the standard method spends 31.00 seconds in the master problems and 9.24 seconds in refinement, whereas standard scenario bundling

Table 4
Average paired comparisons by algorithmic contribution. Entries are class-level means over 200 seeds and are reported as baseline/variant. Times are in seconds and objective gaps are in percent.

Inst.	Compared variants	Time	It.	Blocks	Obj. gap
Panel A. Standard APM vs basis-aware refinement					
4x1600	Standard / BA	2.78 / 3.28	3.45 / 3.79	26.4 / 36.3	10.428 / 0.320
6x2400	Standard / BA	7.34 / 8.39	4.35 / 4.45	188.7 / 214.8	2.739 / 0.078
8x3200	Standard / BA	40.25 / 38.87	4.88 / 4.75	791.4 / 816.3	0.458 / 0.044
Panel B. Standard APM vs scenario bundling					
4x1600	Standard / Std.+SB	2.78 / 3.13	3.45 / 4.00	26.4 / 15.3	10.428 / 10.486
6x2400	Standard / Std.+SB	7.34 / 7.09	4.35 / 4.90	188.7 / 75.2	2.739 / 2.797
8x3200	Standard / Std.+SB	40.25 / 18.06	4.88 / 5.43	791.4 / 334.4	0.458 / 0.504
Panel C. Basis-aware APM vs basis-aware scenario bundling					
4x1600	BA / BA+SB	3.28 / 3.87	3.79 / 4.55	36.3 / 18.9	0.320 / 0.325
6x2400	BA / BA+SB	8.39 / 8.77	4.45 / 5.77	214.8 / 83.3	0.078 / 0.077
8x3200	BA / BA+SB	38.87 / 19.34	4.75 / 5.79	816.3 / 350.6	0.044 / 0.050
Panel D. Basis-aware APM vs K-Means initialization					
4x1600	BA / KM+BA	3.28 / 3.23	3.79 / 2.60	36.3 / 215.5	0.320 / 0.021
6x2400	BA / KM+BA	8.39 / 11.44	4.45 / 2.84	214.8 / 553.0	0.078 / 0.028
8x3200	BA / KM+BA	38.87 / 50.63	4.75 / 2.94	816.3 / 1308.0	0.044 / 0.036
Panel E. Basis-aware APM vs central initialization					
4x1600	BA / Central+BA	3.28 / 3.03	3.79 / 3.45	36.3 / 31.6	0.320 / 0.051
6x2400	BA / Central+BA	8.39 / 6.89	4.45 / 4.00	214.8 / 172.4	0.078 / 0.033
8x3200	BA / Central+BA	38.87 / 27.52	4.75 / 4.30	816.3 / 671.8	0.044 / 0.037

spends 6.71 seconds in the master problems and 11.34 seconds in refinement. K-Means shifts the computational burden in the opposite direction: its 8-item mean consists of 44.78 seconds in the master problems and 5.85 seconds in refinement. This is consistent with its design: the initial clustering gives the algorithm a more informative partition, but it also produces larger master problems. Central initialization is less aggressive, with 19.41 seconds in master problems and 8.10 seconds in refinement for the same class.

4.5. Variant-level behavior

The aggregate tables can also be reorganized into class-level averages that isolate the contribution of each algorithmic modification. Table 4 is organized into five panels. Panel A compares the standard method against basis-aware refinement. Panels B and C compare scenario bundling under the standard and basis-aware refinement criteria, respectively. Panels D and E keep basis-aware refinement fixed and compare it against the two non-trivial initial partitions. Each row reports the mean over the 200 seeds of one instance class for the two variants being compared, using the same seed set on both sides of the comparison.

Panel A shows that basis-aware refinement is primarily an accuracy-oriented modification in average terms. It sharply reduces the mean objective gap in all three classes, from 10.428% to 0.320% in the 4-item class, from 2.739% to 0.078% in the 6-item class, and from 0.458% to 0.044% in the 8-item class. Its time effect is modest for the two smaller classes and favorable for the largest class, where the mean running time decreases from 40.25 to 38.87 seconds. The improvement is consistent with the fact that basis status identifies changes in the follower's active structure rather than small numerical differences in primal solution vectors.

Panels B and C highlight the role of scenario bundling under the two refinement criteria. In every class, bundling reduces the final number of blocks, and the effect is strongest in the largest instances. For the standard method in the 8-item class, the final partition decreases from 791.4 to 334.4 blocks and the mean time decreases from 40.25 to 18.06 seconds, while the objective gap increases from 0.458% to 0.504%. With basis-aware refinement, the same class-level comparison reduces the final partition from 816.3 to 350.6 blocks and the mean time from 38.87 to 19.34 seconds, with the objective gap changing from 0.044% to 0.050%. Thus, bundling is best interpreted as a compact-partition and speed mechanism, especially effective for the largest class and more accurate when combined with basis-aware refinement.

Panels D and E isolate the value of non-trivial initial partitions once basis-aware refinement is already in place. K-Means consistently reduces the mean number of iterations, reaching 2.60, 2.84, and 2.94 iterations across the three classes, and it gives the smallest objective gaps in Table 4. This accuracy comes with larger final partitions and, in the two larger classes, longer mean running times. The central initialization is more balanced: relative to basis-aware refinement alone, it reduces time, iterations, and final partition size in all three classes while keeping the mean objective gap below 0.052%. These paired averages complement Tables 1–3: K-Means is the most accuracy-oriented initialization, while central initialization provides the stronger time–partition-size compromise.

4.6. Discussion

Overall, the experiments support four conclusions. First, adaptive scenario partitioning is an effective alternative to the extended formulation for the stochastic bilevel knapsack instances considered here. All variants provide substantial time savings on average, and the best basis-aware variants maintain objective-value gaps below 0.1% in most runs. Second, the structure of the follower's optimal basis is a

useful refinement signal. It improves exact agreement and sharply reduces the mean objective gap relative to the standard norm-based criterion. Third, informed initial partitions are especially beneficial when high accuracy is required. K-Means with basis-aware refinement is the most accurate variant in the present test bed, whereas central initialization with basis-aware refinement provides a strong compromise between accuracy, time, and partition size. Fourth, scenario bundling is preferable when the main goal is to obtain a fast approximation, especially in combination with basis-aware refinement and in the largest instances, where reducing the number of blocks has the strongest impact on master-problem time.

These results also highlight a practical distinction between matching the exact leader capacity and matching the objective value. Exact agreement becomes less frequent as the instances grow, but objective gaps remain small for the basis-aware variants. For applications in which the leader's realized objective is the primary performance criterion, the tolerance-based results therefore provide a more informative measure of computational quality than exact equality alone. For applications in which reproducing the deterministic-equivalent leader decision is critical, the K-Means and central initializations are the most reliable choices among the tested variants.

5. Conclusions

This work studied the adaptation of the Adaptive Partitioning Method (APM), originally designed for two-stage stochastic programming, to stochastic bilevel linear problems. The motivation stems from the need for efficient approaches capable of addressing the structural and numerical challenges introduced by uncertainty in the follower's parameters.

We began with a standard APM formulation and introduced several methodological enhancements. The first was a refinement rule based on basis-aware information, which groups scenarios according to the optimal basis of the follower problem. This strategy mitigates numerical instability, reduces degeneracy-driven fragmentation and consistently improves convergence accuracy relative to the tolerance-based refinement.

To further reduce iteration counts and avoid excessive growth in scenario blocks, we proposed a Scenario Bundling approach that rebuilds scenario groups from scratch after each master solve. Although this method yields substantial speed-ups, it introduces small deviations from the extended-model optimum, making it attractive when computation time is prioritised over exact accuracy.

We also examined two non-trivial initial partitions: one constructed via K-Means clustering and another (central initialization) based on the follower response evaluated at the center of the leader's feasible range. Both provide the master problem with structurally meaningful scenario blocks from the outset, significantly accelerating convergence without sacrificing solution quality. Across all tested configurations, these initialisation strategies achieved the best balance between precision and efficiency.

Extensive numerical experiments show that the adapted APM is a competitive alternative to solving the extended formulation directly. All variants deliver substantial reductions in computational time, while the basis-aware, K-Means and central-initialization versions consistently recover solutions that match or closely approximate the extended-model optimum. Scenario Bundling further enhances speed, especially in large-scale instances.

Overall, the findings indicate that careful design of both the initial partition and the refinement criterion is crucial for achieving robust and scalable performance in stochastic bilevel problems. Promising directions for future work include adaptive refinement rules based on dual sensitivity, improved clustering heuristics for initial partitions, and the integration of learning-based methods to predict effective partitions or guide refinement. Extending the approach to broader classes of bilevel models and uncertainty structures also represents a natural next step.

References

Beck, Y., Ljubić, I., Schmidt, M., 2023a. Exact methods for discrete Γ-robust interdiction problems with an application to the bilevel knapsack problem. Mathematical Programming Computation 15, 293–337. URL: https://doi.org/10.1007/s12532-023-00244-6, doi:10.1007/s12532-023-00244-6.

Beck, Y., Ljubić, I., Schmidt, M., 2023b. A survey on bilevel optimization under uncertainty. European Journal of Operational Research 311, 401–426. URL: https://doi.org/10.1016/j.ejor.2023.01.008, doi:10.1016/j.ejor.2023.01.008.

Beck, Y., Schmidt, M., 2021. A gentle and incomplete introduction to bilevel optimization. Optimization Online URL: https://optimization-online.org/?p=17182.

Besançon, M., Anjos, M.F., Brotcorne, L., 2024. Robust bilevel optimization for near-optimal lower-level solutions. Journal of Global Optimization 89, 107–126. URL: https://doi.org/10.1007/s10898-024-01422-z, doi:10.1007/s10898-024-01422-z.

Bounitsis, D., Papageorgiou, L.G., Charitopoulos, V.M., 2022. Data-driven scenario generation for two-stage stochastic programming. Chemical Engineering Research and Design 187, 206–224. URL: https://doi.org/10.1016/j.cherd.2022.08.032, doi:10.1016/j.cherd.2022.08.032.

Buchheim, C., Henke, D., Irmai, J., 2022. The stochastic bilevel continuous knapsack problem with uncertain follower's objective. Journal of Optimization Theory and Applications 194, 521–542. URL: https://doi.org/10.1007/s10957-022-02037-8, doi:10.1007/s10957-022-02037-8.

Camacho-Vallejo, J.F., Corpus, C., Villegas, J.G., 2024. Metaheuristics for bilevel optimization: A comprehensive review. Computers & Operations Research 161, 106410. URL: https://doi.org/10.1016/j.cor.2023.106410, doi:10.1016/j.cor.2023.106410.

Chen, L., Xiao, S., Balasubramanian, K., 2024. Optimal algorithms for stochastic bilevel optimization under relaxed smoothness conditions. Journal of Machine Learning Research 25, 1–51. URL: https://www.jmlr.org/papers/v25/23-1323.html.

Claus, M., 2021. On continuity in risk-averse bilevel stochastic linear programming with random lower level objective function. Operations Research Letters 49, 511–515. URL: https://doi.org/10.1016/j.orl.2021.05.006, doi:10.1016/j.orl.2021.05.006.

Claus, M., 2022. Existence of solutions for a class of bilevel stochastic linear programs. European Journal of Operational Research 299, 542–549. URL: https://doi.org/10.1016/j.ejor.2021.09.035, doi:10.1016/j.ejor.2021.09.035.

Dempe, S., 2002. Foundations of Bilevel Programming. Nonconvex Optimization and Its Applications, Springer Science & Business Media. URL: https://doi.org/10.1007/b101970, doi:10.1007/b101970.

Fairbrother, J., Turner, M., Wallace, S.W., 2022. Problem-driven scenario generation: an analytical approach for stochastic programs with tail risk measure. Mathematical Programming 191, 141–182. URL: https://doi.org/10.1007/s10107-019-01451-7, doi:10.1007/s10107-019-01451-7.

Forcier, F., Gaubert, S., Leclère, V., 2024. Exact quantization of multistage stochastic linear problems. SIAM Journal on Optimization 34, 1857–1885. URL: https://doi.org/10.1137/22M1508005, doi:10.1137/22M1508005.

Fortuny-Amat, J., McCarl, B., 1981. A representation and economic interpretation of a two-level programming problem. The Journal of the Operational Research Society 32, 783–792. URL: http://www.jstor.org/stable/2581394, doi:10.2307/2581394.

García-Vélez, J.C., Ruiz-Hernández, D., Camacho-Vallejo, J.F., Díaz, J.A., 2024. Competitive network restructuring with spatially loyal customers. a bilevel facility delocation problem. Computers & Operations Research 161, 106409. URL: https://doi.org/10.1016/j.cor.2023.106409, doi:10.1016/j.cor.2023.106409.

Kermani, M., Cordeau, J.F., Jans, R., 2024. A progressive hedging-based matheuristic for the stochastic production routing problem with adaptive routing. Computers & Operations Research 169, 106745. URL: https://doi.org/10.1016/j.cor.2024.106745, doi:10.1016/j.cor.2024.106745.

Kleinert, T., Labbé, M., Ljubić, I., Schmidt, M., 2021. A survey on mixed-integer programming techniques in bilevel optimization. EURO Journal on Computational Optimization 9, 100007. URL: https://doi.org/10.1016/j.ejco.2021.100007, doi:10.1016/j.ejco.2021.100007.

Lozano, L., Borrero, J.S., 2024. A bilevel optimization approach for a class of combinatorial problems with disruptions and probing. INFORMS Journal on Computing 36, 1662–1677. URL: https://doi.org/10.1287/ijoc.2024.0629, doi:10.1287/ijoc.2024.0629.

Mahéo, A., Belières, S., Adulyasak, Y., Cordeau, J.F., 2024. Unified branch-and-benders-cut for two-stage stochastic mixed-integer programs. Computers & Operations Research 164, 106526. URL: https://doi.org/10.1016/j.cor.2023.106526, doi:10.1016/j.cor.2023.106526.

Muñoz, G., Salas, D., Svensson, A., 2025. Exploiting the polyhedral geometry of stochastic linear bilevel programming. Mathematical Programming 210, 695–730. URL: https://doi.org/10.1007/s10107-024-02097-w, doi:10.1007/s10107-024-02097-w.

Pinzon Ulloa, D., Frejinger, E., Gendron, B., 2024. A logistics provider's profit maximization facility location problem with random utility maximizing followers. Computers & Operations Research 167, 106649. URL: https://doi.org/10.1016/j.cor.2024.106649, doi:10.1016/j.cor.2024.106649.

Rahmaniani, R., Crainic, T.G., Gendreau, M., Rei, W., 2024. An asynchronous parallel benders decomposition method for stochastic network design problems. Computers & Operations Research 162, 106459. URL: https://doi.org/10.1016/j.cor.2023.106459, doi:10.1016/j.cor.2023.106459.

Ramirez-Pico, C., Moreno, E., 2022. Generalized adaptive partition-based method for two-stage stochastic linear programs with fixed recourse. Mathematical Programming 196, 755–774. URL: https://doi.org/10.1007/s10107-020-01609-8, doi:10.1007/s10107-020-01609-8.

Roland, M., Forel, A., Vidal, T., 2025. Adaptive partitioning for chance-constrained problems with finite support. SIAM Journal on Optimization 35, 476–505. URL: https://doi.org/10.1137/24M1632772, doi:10.1137/24M1632772, arXiv:https://doi.org/10.1137/24M1632772.

Salo, A., Andelmin, J., Oliveira, F., 2022. Decision programming for mixed-integer multi-stage optimization under uncertainty. European Journal of Operational Research 299, 550–565. URL: https://doi.org/10.1016/j.ejor.2021.12.013, doi:10.1016/j.ejor.2021.12.013.

Shapiro, A., Dentcheva, D., Ruszczyński, A., 2009. Lectures on stochastic programming. Modeling and theory. doi:10.1137/1.9780898718751.

Song, Y., Luedtke, J., 2015. An adaptive partition-based approach for solving two-stage stochastic programs with fixed recourse. SIAM Journal on Optimization 25, 1344–1367. URL: https://doi.org/10.1137/140967337, doi:10.1137/140967337.

von Stackelberg, H., 1934. Marktform und Gleichgewicht. J. Springer.

Zhang, Y., Mazzi, N., McKinnon, K., Garcia Nava, R., Tomasgard, A., 2024. A stabilised benders decomposition with adaptive oracles for large-scale stochastic programming with short-term and long-term uncertainty. Computers & Operations Research 167, 106665. URL: https://doi.org/10.1016/j.cor.2024.106665, doi:10.1016/j.cor.2024.106665.